



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/559,849	04/27/2000	Albhy Galuten	9386/1F566US1	2145

7590 09/25/2003  
Darby & Darby PC  
805 Third Avenue  
New York, NY 10022

EXAMINER

CHU, GABRIEL L

ART UNIT	PAPER NUMBER
----------	--------------

2184

DATE MAILED: 09/25/2003

10

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/559,849

Applicant(s)

GALUTEN ET AL.

Examiner

Gabriel L. Chu

Art Unit

2184

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 24 July 2003.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 2-6 and 8-42 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 32 is/are allowed.
- 6) ☒ Claim(s) 2-6, 8-24, 26-31 and 33-42 is/are rejected.
- 7) ☒ Claim(s) 25 is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- 11) ☐ The proposed drawing correction filed on \_\_\_\_\_ is: a) ☐ approved b) ☐ disapproved by the Examiner.
- If approved, corrected drawings are required in reply to this Office action.
- 12) ☐ The oath or declaration is objected to by the Examiner.

**Priority under 35 U.S.C. §§ 119 and 120**

- 13) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.
- 14) ☒ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).
- a) ☐ The translation of the foreign language provisional application has been received.
- 15) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

**Attachment(s)**

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413) Paper No(s). \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

**DETAILED ACTION**

***Claim Objections***

1. Claim 34 is objected to because of the following informalities: "the substantially" is understood to refer to "substantially", correcting a grammar error. Appropriate correction is required.

***Claim Rejections - 35 USC § 102***

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

3. Claims 2, 4-6, 8, 12-15, 17, 22, 24, 26, 33, 34, 36, 38, 39, and 41 are rejected under 35 U.S.C. 102(a) as being anticipated by US 6000046 to Passmore. Referring to claim 5, Passmore discloses utilizing an error detection system to intercept an error event from one of a plurality of applications (From the abstract, "A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file."); propagating appropriate error alerts to relevant subsystems (From the abstract, "The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error."); filtering errors that

require different levels of response (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible."); and resolving the error event (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.").

Referring to claim 2, Passmore discloses the resolving step includes the further steps of selecting and dispatching appropriate help information to the user (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.").

Referring to claims 4, 12, and 22, Passmore discloses prioritizing errors when there is more than one error still unresolved at any given time (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition.").

Referring to claims 6 and 14, Passmore discloses directing errors to resources capable of assisting in resolving the error (From the abstract, "A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file.").

Referring to claim 13, Passmore discloses utilizing an error detection system to intercept an error event from one of a plurality of system elements (From the abstract,

"A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file."); creating an informative error package related to the error event (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user."); propagating appropriate error alerts to one or more relevant system elements (From the abstract, "The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error."); filtering the error events that require different responses (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message

Art Unit: 2184

first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible."); and resolving the error event using the information contained within the informative error package and resources available within said distributed computer system (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.").

Referring to claims 8 and 17, Passmore discloses the resolving step includes the further step of dispatching appropriate help information to the system element from which the error event originated (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block

506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

Referring to claim 15, Passmore discloses receiving an error message at a central-resource from the system element indicating the occurrence of an error associated with the system element (From the abstract, “A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file.”); referencing an error-resource having a plurality of assistance options (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); selecting an assistance option from the plurality of assistance options in accordance with the error message (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); and providing the assistance option to the system element substantially immediately following the receiving step (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message



sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

Referring to claim 24, Passmore discloses propagating an error alert to one or more components of said computer system that may be affected by the occurrence of said error (From the abstract, “The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error.”).

Referring to claim 26, Passmore discloses receiving a signal indicating an error associated with one of the system elements (From the abstract, “A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file.”); dispatching assistance to the system element associated with the error (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); propagating an error alert message to

Art Unit: 2184

one or more other system elements that may be affected by the error (From the abstract, "The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error."); prioritizing the error relative to other errors not yet resolved (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition."); applying an error filter to the error to determine one or more appropriate responses to the error (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of

the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible.”); selecting one or more assistance options from the one or more appropriate responses in order to resolve the error (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.” Further, from line 27 of column 1, “One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error

may be buried within the user screen and not easily visible."); and resolving the error according to the selected assistance option (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.").

Referring to claim 33, Passmore discloses utilizing an error detection system to intercept an error event from one of a plurality of system elements (From the abstract, "A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file."); creating an error package related to the error event (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user."); filtering errors that require different levels of response (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are

Art Unit: 2184

displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible."); propagating appropriate error alerts to one or more relevant system elements (From the abstract, "The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error."), and resolving the error event using information contained within the error package and resources available within the distributed computer system (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the

message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

Referring to claim 34, Passmore discloses receiving an error message from the system element indicating the occurrence of an error associated with the system element (From the abstract, “A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file.”); referencing an error-resource having a plurality of assistance options (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); selecting an assistance option from the plurality of assistance options in accordance with the error message (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); and providing the assistance option to the system element substantially immediately following the receiving step (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the

error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

Referring to claims 36 and 41, Passmore discloses receiving a signal indicating an error associated with one of the system elements (From the abstract, “A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file.”); dispatching assistance to said system element associated with said error (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); propagating an error alert message to one or more other system elements that may be affected by said error (From the abstract, “The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error.”); prioritizing said error relative to other errors not yet resolved (From line 49 of column 5, “Because the error processor of the flowchart of FIG. 5 is a separate process

within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition.”); applying an error filter to said error to determine one or more appropriate responses to said error (From line 49 of column 5, “Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition.” Further, from line 27 of column 1, “One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible.”); selecting one or



more assistance options from the one or more appropriate responses in order to resolve said error (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible."); and resolving said error according to said selected assistance option (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.").

Referring to claim 38, Passmore discloses utilizing an error detection system to intercept an error event from one of a plurality of system elements (From the abstract, "A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages, and display the help file."); creating an error package (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user."); filter errors that require a different level of response (From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that

displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible.”); propagating appropriate error alerts to one or more relevant system elements (From the abstract, “The common error process detects whether the system is a distributed application running on multiple computer systems, and if this is so, the common error process sends any error messages to other computers within the distributed network, so that the error messages are displayed on all computers when one computer has an error.”), and resolving the error event using the information contained within the informative error package and resources available within said distributed computer system (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

Referring to claim 39, Passmore discloses receiving an error message from said system element indicating the occurrence of an error associated with a system element (From the abstract, “A system uses a common error processing process within a computer system wherein other processes that detect errors send an error message to the common process and the common process is used to display all error messages,

Art Unit: 2184

and display the help file.”); referencing an error-resource having a plurality of assistance options (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); selecting an assistance option from said plurality of assistance options in accordance with said error message (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”); and providing said assistance option to said system element substantially immediately following said receiving step (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”).

***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 3, 11, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6000046 to Passmore as applied to claims 5, 13, and 26 above. Referring to claims 3, 11, and 27, although Passmore does not specifically disclose locating an alternative resource to substitute for a failed resource associated with the intercepted error, this is well known in the art. Examiner takes official notice for an alternative resource for a failed resource. An example of this is failover. A person of ordinary skill in the art at the time of the invention would have been motivated to implement failover in a system because being able to fail over to a similar resource in the event of failure allows a system to be more resilient.

6. Claims 9, 10, 16, 18, 21, 23, 28, 29, and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6000046 to Passmore as applied to claims 13, 15, and 26 above, and further in view of US 5666481 to Lewis. Referring to claims 9, 18, and 28, although Passmore does not specifically disclose dynamically updating the help information, updating the help information is well known in the art. An example of this is shown by Lewis, from the abstract, "Completed trouble tickets are stored in a library and when an outstanding trouble ticket is received, the system uses at least one determinator to correlate the outstanding communications network fault to data fields in the set of data fields of the trouble ticket data structure to determine which completed trouble tickets in the library are relevant to the outstanding communications network fault." A person of ordinary skill in the art would have been motivated to update help

information in a fault resolution system because, from the abstract, "the system uses at least one determinator to correlate the outstanding communications network fault to data fields in the set of data fields of the trouble ticket data structure to determine which completed trouble tickets in the library are relevant to the outstanding communications network fault".

Referring to claims 10, 16, and 29, Passmore discloses receiving a query for help information from one of said plurality of system elements (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212."); and providing help information in response to the help query (From line 29 of column 5, "Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user."). Although Passmore does not specifically disclose dynamically updating the help information, updating the help information is well known in the art. An example of this is shown by Lewis, from the abstract, "Completed trouble tickets are stored in a library and when an outstanding trouble ticket is received, the system uses at least one determinator to correlate the outstanding communications network fault to data fields in the set of data fields of the trouble ticket data structure to determine which completed trouble tickets in the library are relevant to the outstanding communications network fault." A person of ordinary skill in the art would have been motivated to update help information in a fault resolution system because, from the abstract, "the system uses at least one determinator to

correlate the outstanding communications network fault to data fields in the set of data fields of the trouble ticket data structure to determine which completed trouble tickets in the library are relevant to the outstanding communications network fault”.

Referring to claim 21, Passmore discloses a help file for the resolution of errors (From line 27 of column 5, “Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user.”). Although Passmore does not specifically disclose said error-resource comprises a database containing information regarding bypassing of errors, having a help file present information for bypassing errors is well known in the art. An example of this is shown by Lewis, from the abstract, “An improved method and apparatus of resolving faults in a communications network. The preferred system uses a trouble ticket data structure to describe communications network faults. Completed trouble tickets are stored in a library and when an outstanding trouble ticket is received, the system uses at least one determinator to correlate the outstanding communications network fault to data fields in the set of data fields of the trouble ticket data structure to determine which completed trouble tickets in the library are relevant to the outstanding communications network fault. The system retrieves a set of completed trouble tickets from the library that are similar to the outstanding trouble ticket and uses at least a portion of the resolution from at least one completed trouble ticket to provide a resolution of the outstanding trouble

ticket. The determinators may be macros, rules, a decision tree derived from an information theoretic induction algorithm and/or a neural network memory derived from a neural network learning algorithm. The system may adapt the resolution from a retrieved trouble ticket to provide the resolution using null adaptation, parameterized adaptation, abstraction/respecialization adaptation, or critic-based adaptation techniques. " A person of ordinary skill in the art would have been motivated to use a fault resolution database when presented with an error in the system because, from line 20 of column 3, 'Still another object of the present invention is to provide a method and apparatus for automatically resolving faults in communications networks.'

Referring to claims 23 and 31, Passmore discloses generating an information-package in response to the receiving step (From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user."). Although Passmore does not specifically disclose what the message can contain, Lewis discloses an error message comprising an error-identification (From figure 3, element 62A, 62B, or 62L.), and at least one of a system identification (From figure 3, element 62D, 62E, 62G, or 62J.), an application identification (From figure 3, element 62D, 62E, 62G, or 62J), a time stamp (From figure 3, element 62D, 62H, or 62K.), a location (From figure 3, element 62D or 62G.), a priority (From figure 3, element 62S or 62U.), and an internal state (From figure 3, any



element in the figure is indicative of or related to state and is internal to the system.). A person of ordinary skill in the art at the time of the invention would have been motivated to use a trouble ticket because from line 20 of column 3, 'Still another object of the present invention is to provide a method and apparatus for automatically resolving faults in communications networks.'

7. Claims 19 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6000046 to Passmore as applied to claims 17 and 26 above, and further in view of US 5983364 to Bortcosh et al. Referring to claims 19 and 30, although Passmore does not specifically disclose investigating the error by soliciting additional information about the occurrence of the error from the system element associated with the error, soliciting additional information is well known in the art. An example of this is shown by Bortcosh et al., from the abstract, "An expert system unit analyzes the fault and uses the file set to determine a fact request to be asked or a remedy to be applied to the computer system." A person of ordinary skill in the art at the time of the invention would have been motivated to perform a fact request because, from the abstract, "The expert system makes additional fact requests if needed until a diagnosis is reached or the current file set is exhausted."

8. Claims 20, 35, 37, 40, and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6000046 to Passmore as applied to claims 15, 34, 36, 39, and 41 above, and further in view of US 5748880 to Ito et al. Referring to claims 20, 35, 37, 40, and 42, although Passmore does not specifically disclose determining a preferred assistance option based on previous assistance options provided in response to an

Art Unit: 2184

error; and wherein the selecting step includes selecting the preferred assistance option, selecting a past resolution is known in the art. An example is shown by Ito et al. from the abstract, "The system further includes a recovery history-storing device for storing recovery procedures previously executed by the recovery-executing device; a recovery history-selecting device for selecting one of the recovery procedures stored in the recovery history-storing device, according to an operation of the operator; and a recovery history display device for displaying the selected one recovery procedure on the display screen." Further, from line 66 of column 5, "The menu indication area 51g displays a menu of functions utilized in recovery from the fault. If "LOG" is selected by the cursor 50d, a log (recovery history) of the preceding recovery operation corresponding to the selected error message is read from the hard disk drive 15, and displayed on the recovery procedure indication area 51e, as shown in FIG. 7, as a window of a recovery log display screen 52. The operator performs operations according to indications on the recovery procedure indication area 51e or the recovery log display screen 52, to execute recovery from the fault." A person of ordinary skill in the art at the time of the invention would have been motivated to use a resolution history in determining a current resolution because, from line 8 of column 2, it is "the object thereof is to provide a computer-supervising system which makes it possible to confirm a type of a fault, and effect recovery from the fault, more promptly."

***Allowable Subject Matter***

9. Claim 25 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the

base claim and any intervening claims. Referring to claim 25, the prior art does not teach or fairly suggest, in light of the parent claim, said receiving step further comprises the steps of determining whether a connection exists between said system element and said central-resource; transmitting said error message from said system element to said central resource when said connection exists; and when said connection does not exist, queuing said error message for later transmission to said central resource.

10. Claim 32 is allowed.

11. The following is an examiner's statement of reasons for allowance: Referring to claim 32, the prior art does not teach or fairly suggest a filter for sorting errors of different error types and directing said errors through a routing server to different components of a distributed computer system in accordance with the error type, in a system for tracking and processing errors that occur in a distributed computer system, wherein the system comprises: a resource server for processing the errors; a database of information accessible to the resource server and useful in resolving and bypassing said errors; and a routing server for contemporaneously directing messages and responses to said errors to and from components of said distributed computer system.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

***Response to Arguments***

12. Applicant's arguments filed 24 July 2003 have been fully considered but they are not persuasive. Referring to Applicant's argument that Passmore does not filter errors by the specification's description of "filtering", the specification's description of filtering was not claimed. From Applicant's arguments, "Claims 5, 13, 26, 36, and 41 require the step of filtering the errors to sort which errors require different responses." Filtering in this case is broadly and reasonably interpreted as passing through a filter. An example of a filter is an all-pass filter, a filter in which everything is passed. The claims as written do not distinctly point out what action, if any, the filter performs. Applicant further argues that Passmore does not filter errors, however the system disclosed by Passmore operates as a filter at least inasmuch as an all-pass filter with respect to the display of errors. However, Passmore goes further to select an error, the first, from all the errors for topmost display. From line 49 of column 5, "Because the error processor of the flowchart of FIG. 5 is a separate process within the computer system, and all error messages are transferred to the error processor for display, there will not be any priority conflict as to which messages are displayed first. Thus, since all error messages are displayed by the error processor, the first error message received will be displayed first, and this error message is most likely the error message produced by the process that first discovered the error condition." Further, from line 27 of column 1, "One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the

computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible.”

Referring to Applicant’s argument that Passmore does not prioritize errors by the specification’s description of “prioritizing”, the specification’s description of prioritizing was not claimed. Prioritizing in this case is broadly and reasonably interpreted as arranging to deal with in order of importance. From line 27 of column 1, “One important aspect of error processing is to attempt to produce the earlier error message first within the system. That is, the first process to detect an error more often finds the actual, or root, cause of the error, and it is important that the error message displayed by this first process be the first or top most error message on the display screen connected to the computer system that displays the error. This may not occur if multiple processes detect the error, since some of those processes may have a higher priority than the process that first detected the error, and the process with the higher priority will display its message before the other processes. In this situation, the message from the process that first detects the error may be buried within the user screen and not easily visible.” Clearly, Passmore places the emphasis on chronological order, indicating that that is what is “important”. Further, while the emphasis is placed on chronological order, Passmore further places emphasis on the priority of processes.

Referring to Applicant's argument that Passmore does not disclose dispatching assistance to a system element, wherein a system element is a "broad range of computer programs and sub-systems ... [including], for example, applications programs, sub-programs, operating systems communication protocols, and drivers for peripherals", this description is not claimed, and further, is intentionally vague, using terms like "broad" and "for example". A system element is an element in a system. A system, furthermore, is a group of interacting, interrelated, or interdependent elements forming a complex whole. In this case, a user process executes on a computer, and an error message is transmitted to every computer on the network, one such every computer being the computer on which the process which detected the error executed. From line 27 of column 5, "Referring now to FIG. 5, after entry, block 502 retrieves the error message sent by the error process 210 or 212. Block 504 then formats this error message for display to the user, block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208 (FIG. 2), where the message is displayed to the user." What "appropriate help information" is is further unclaimed, however, again, "block 505 finds a reference in the help system for the message and inserts it into the message, and block 506 sends the formatted message to the error GUI 208".

Referring to Applicant's argument that Lewis does not prioritize or filter errors, these points are addressed by Passmore, as indicated above.

Referring to Applicant's argument that Bortcosh does not disclose how multiple problems are handled, Passmore addresses this point, as indicated above.

Referring to Applicant's argument that Ito does not prioritize or transmit assistance, these points are addressed by Passmore, as indicated above.

***Conclusion***

13. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Gabriel L. Chu whose telephone number is (703) 308-7298. The examiner can normally be reached on weekdays with alternate Fridays off.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert W. Beausoliel, Jr. can be reached on (703) 305-9713. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Application/Control Number: 09/559,849  
Art Unit: 2184

Page 31

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

gc

  
ROBERT BEAUSOLIEL  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100